DESIGN OF A RUGGED LOCAL NETWORK FOR DISTRIBUTED CONTROL SYSTEMS AND OFFICE AUTOMATION

Bahman Zargham

Cromemco, Inc.
280 Bernardo Avenue
Mountain View, California

INTRODUCTION

In the decade since the first microprocessor was introduced, microcomputers have come to be used in almost all aspects of computing. A simple rule has governed the evolution of the microcomputer world: the more powerful and less expensive the computer, the more decentralized the computing power of an office or a factory becomes.

Microprocessors were first used as control nodes, which needed a way of communicating with the outside world. Terminals, printers, and disks were added, and the result was today's microcomputers. The fact that the CPU no longer accounted for most of the computer's cost changed the whole problem of computer efficiency. Everybody could afford a CPU, but it was not cost effective for everybody to have a hard disk and fast printer.

CROMIX OPERATING SYSTEM

Multi-user, Multi-process System

As the microcomputer world grew, more sophisticated software for microcomputers developed. It became obvious that people working in the same office needed to share information more rapidly. Passing diskettes among users was inefficient. Also, the cost of expensive electromechanical devices such as hard disks and large printers was not decreasing as fast as the cost of microprocessors. Therefore, such peripherals had to be shared among several users.

Cromemco's solution to this problem was to develop a multi-user, multi-process operating system called Cromix (similar to Unix). This time-sharing disk operating system allowed many users to share one single CPU and peripherals. The performance of such a system is satisfactory as long as demand on the central processor is reasonable.

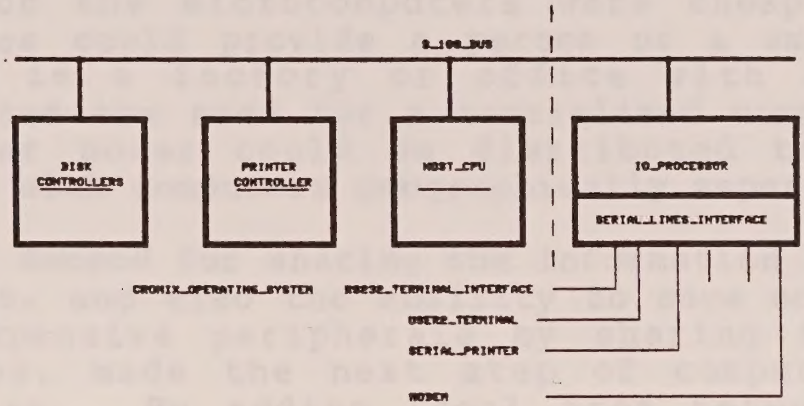

Figure 1: CROMIX SYSTEM

Multi-processor, Multi-user Environment

The next step was to give the people who needed more speed their own processor, while still sharing the peripherals and keeping the same level of data communication between users. Figure 2 shows a Cromix system sharing the disk between single-user machines. This environment brought about the required speed and sharing of information.

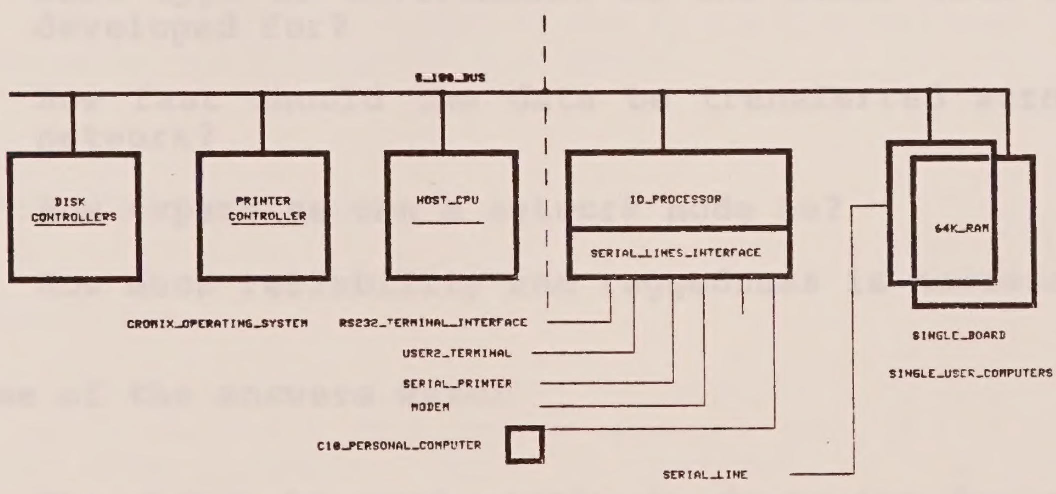

Figure 2: MULTIPROCESSOR ENVIRONMENT

However, bringing more data-processing power to the user created a need to transfer this processed information faster and farther between machines, so that it could be accessed by users in different departments. In this case, RS-232 serial lines are not fast enough and cannot

support the necessary distances. Also, they do not have the bandwidth needed for many network applications.

LOCAL AREA NETWORK

The fact the microcomputers were cheap enough that companies could provide a person or a small group of people in a factory or office with a computer, eliminated the need for a centralized computer center. Computer power could be distributed throughout an office, with computers geographically separated.

Growing demand for sharing the information between these machines, and also the ablility to save on the cost of very expensive peripherals by sharing them between machines, made the next step of computer evolution necessary. By adding local area networks, it was possible to make the computing power of an office even more reliable and more distributed. The information would be instantly available to everyone who needed it, as long as they were authorized to access it.

Cromemco Local Area Network (C-Net)

After deciding that a local area network should be developed, there were a few questions that had to be answered. Some of these questions were:

1. What type of environment is the local area network developed for?

2. How fast should the data be transferred across the network?

3. How expensive can a network node be?

4. How much reliability and ruggedness is necessary?

Some of the answers were:

1. The C-Net is used mostly in factories for process control or distributed monitoring systems, with medium-range data transfer rate requirements.

2. Since the network is designed for a microcomputer world, the speed of data transfer should be in proportion with the speed of the microprocessors that will process the data.

3. Microcomputers are not very expensive. Therefore, the network node cost should be reasonable compared with the cost of the whole machine. Off-the-shelf communication chips should be used wherever possible.

4. The network should be rugged so it can survive environments other than offices.

A data transfer rate of around 500 kbits per second is adequate, being in proportion to microprocessor speeds. The question of which topology to choose for the network was easily answered. A flexible configuration was needed, so it would be possible to conveniently route the network through offices and factories with different geometries, and to reduce the cost of installation. In this respect, an open-bus topology was chosen.

For ruggedness, the physical link and network interface were based on military standard 1553B for aircraft internal buses. The coupling of network nodes to the transmission media was done through an isolation transformer (known as AC coupling). Differential amplifiers were then used for transmitting and receiving. These two factors improved ruggedness, reliability and noise reduction of the whole network greatly.

Because of the transformer, the computer and network node electronics are completely isolated from the cable. This makes them immune to faults in the network cable because of lightning or electrical shorts. For example, 110 VAC has been applied directly to the network in lab tests with no damage to the nodes.

C-NET PHYSICAL LAYER

C-Net Cable

A shielded, twin-axial cable (Belden 9272 or equivalent) is used for both the C-Net network bus cable and the C-Net drop cables. This is inexpensive and provides better noise immunity than either twisted pair or coaxial cable.

C-Net Signals

C-Net specifications require a digitally phase-encoded modulation of signal at 500 KHZ, +/-0.001%. The station transmitter must provide a differential signal of 5

volts +/-10% peak-to-peak at the output winding of the coupling transformer.

CROMEMCO NETWORK INTERFACE (CNI)

The CNI board is the interface between the network cable and the computer I/O processor (IOP). The IOP is an S-100 bus-resident single-board computer, featuring a Z80A processor, 16K to 64K bytes of RAM program storage, and socket space for up to 32K bytes of ROM.

All CNI functions are controlled by the program running in IOP memory.

CNI Block Diagram

The CNI board is partitioned by function, as follows:

1. C-Net/Altmode select
2. C-Net coupling transformer and transceiver
3. Network status and control
4. Network phase encoder
5. Network clock recovery and phase decoder
6. Network UART
7. General-purpose timer

CNI Transformer and Transceiver

All the station transceivers are transformer-coupled to the C-Net cable. Transformer coupling and differential amplifiers provide superior fault isolation, and high common mode noise rejection. The coupling capacitors between the isolation transformer and the network cable serve the following functions:

1. They present a very high impedance to 60 HZ common mode noise and a low impedance to C-Net signals.
2. They allow C-Net cable to be tested for short circuits with an ordinary ohmmeter.

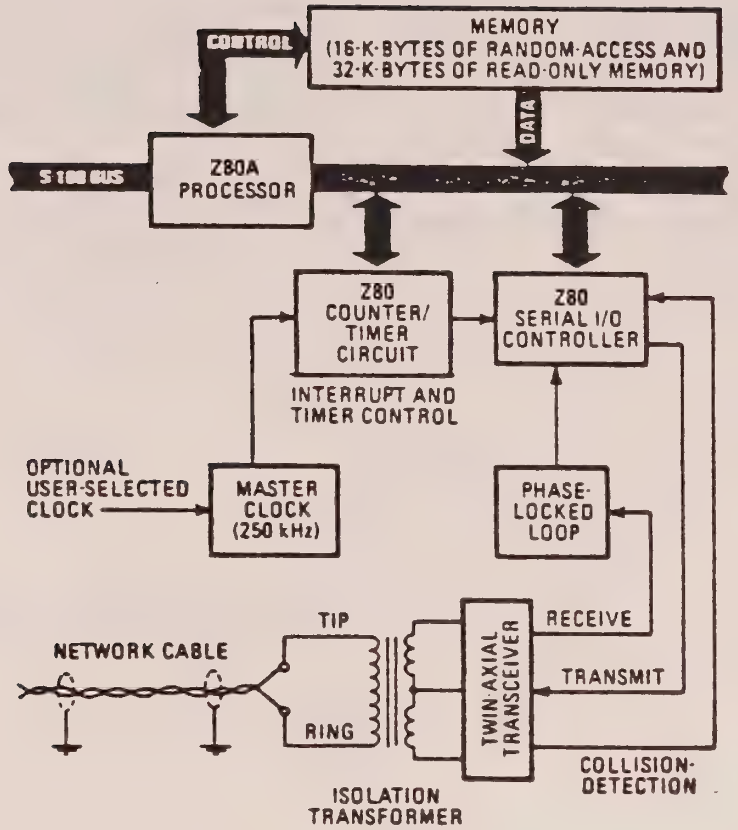

Figure 3: CNI BLOCK DIAGRAM

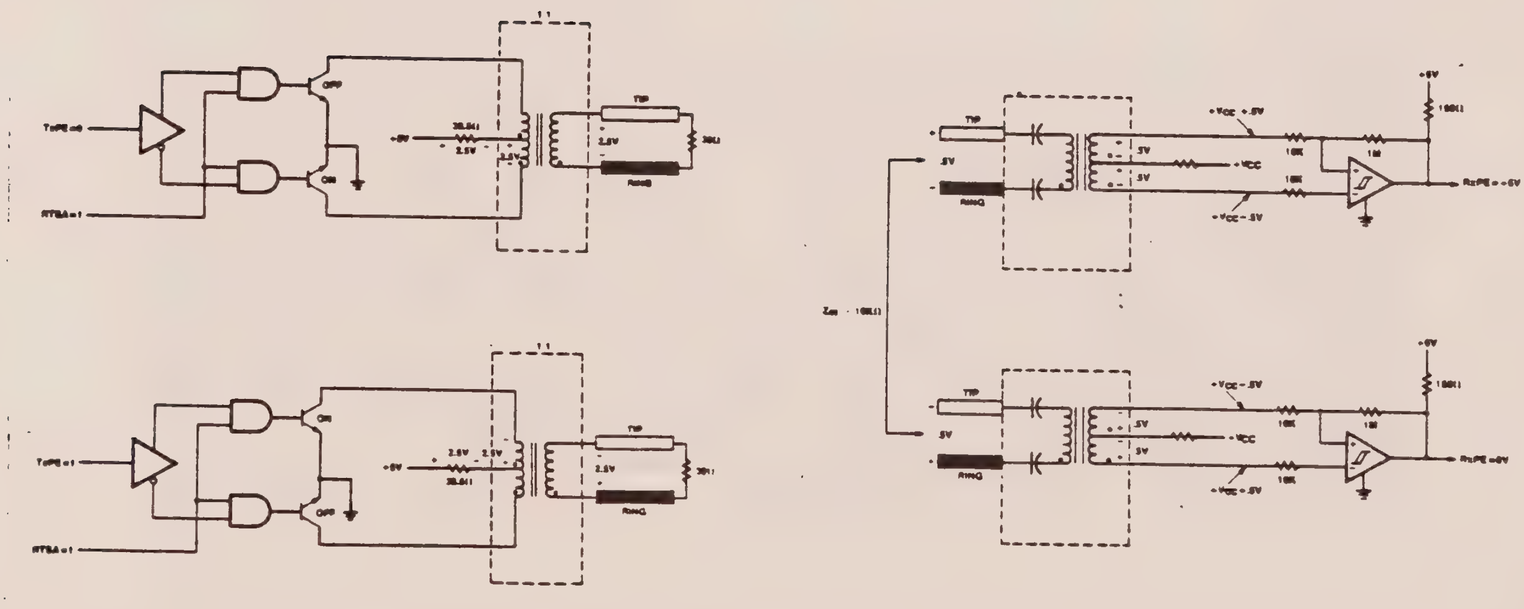

Figure 4: CNI TRANSMITTER AND CNI RECEIVER

CNI Carrier Detection

The CNI carrier detection circuitry is shown in Figure 5. It has a very simple design. Two single shots alternately fire on rising and falling edges of the phase-encoded signal output of the network receiver in C-Net mode. The idle line remains low, provided there is at least one RxPE edge during two full bit intervals. Otherwise, the idle line goes high, indicating no carrier on the cable.

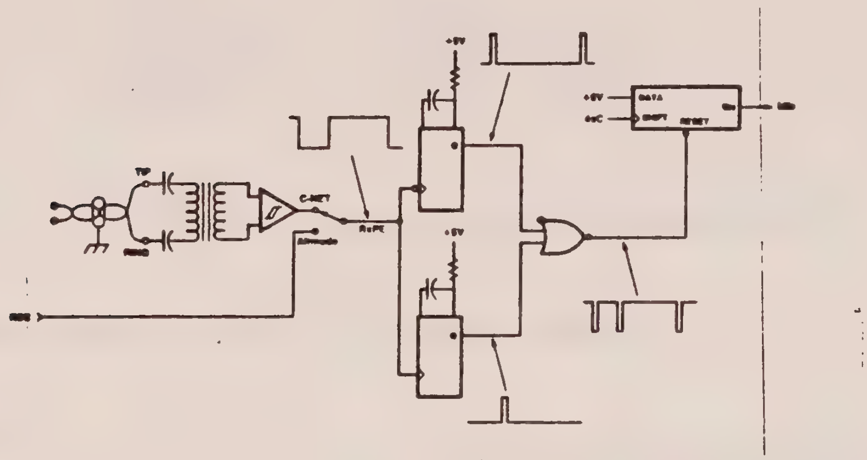

Figure 5: CNI CARRIER DETECTION

CNI Collision Detection

Collision detecting for an AC coupled network was the most difficult part of the network interface board. For this reason, Cromemco has a patent pending on this procedure. The CNI senses a C-Net transmitter collision by monitoring the current flowing into the isolation transformer center tap. The current is sampled by monitoring the center tap voltage VCT. When a CNI is the only active C-Net transmitter (no collision condition), the DC component of VCT is constant. When one CNI transmitter collides with another active C-Net transmitter, the DC component of VCT varies. The variation of the DC component depends on the amplitudes and relative phase of the colliding signals. See Figure 6.

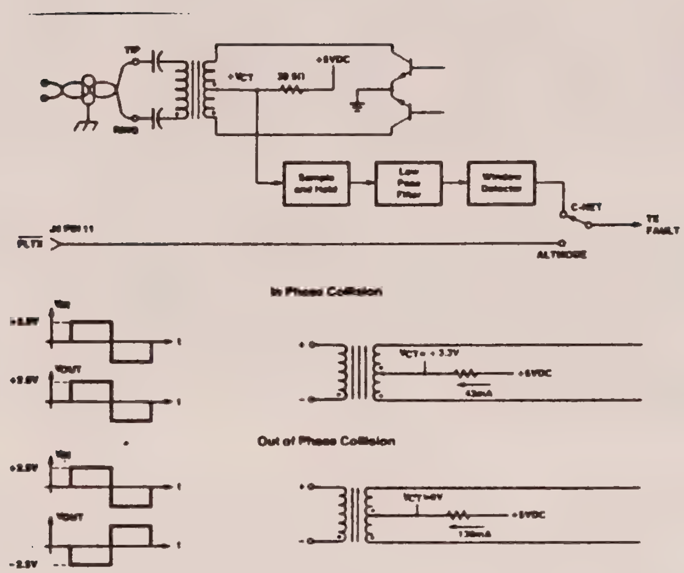

Figure 6: CNI COLLISION DETECTION

Specifications of CNI Physical Layer

Data rate:	500 kbits/s with C-Net 880 kbits/s in Altmode
Network topology:	open bus
Max. station separation:	2000 meter
Max. stations per segment:	255
Medium:	shielded twin-axial cable
Driver type:	differential
Transmission:	binary phase shift coding
Media access procedure:	carrier sense multiple access with collision detection (CSMA/CD)
Coupling to the cable:	transformer coupled/isolated

DATA LINK LAYER

The major task for this layer is managing the transmission of packets of data to other machines, listening to the line for receiving packets from other machines, and giving them to higher layers. In this layer, the network listens to the lines when sending to

see if any other machine is using it. If not, it starts transmitting the packet. Otherwise, it waits until the line is idle. If some other machine happens to start transmitting at the same time, the collision is sensed and the transmission is rescheduled.

Although this layer guarantees the transmission of a packet (or at least a report of excessive collisions, if applicable), it does not guarantee that the transmitted packet will be received by the targeted machine (i.e., this layer of the network sends the data as datagrams to a specific machine, or it can broadcast it to all machines).

If this layer is not transmitting, it is listening for packets addressed to this station and general broadcasts. When receiving, if the first address byte of the packet traveling on the network matches the first byte of the station address, the packet is automatically picked up from the cable by the hardware. Then the next five address bytes are checked. This layer picks up the packets that are received correctly (no CRC error) and passes them up to higher layers. The link layer packet structure is shown in Figure 7.

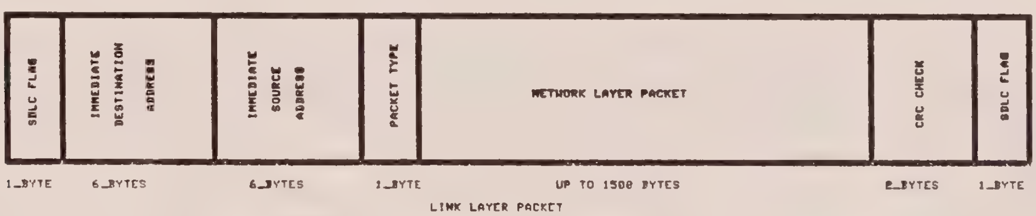

Figure 7: LINK LAYER PACKET

NETWORK LAYER

This layer of the C-Net uses a simple routing algorithm to deliver packets to different sections of a C-Net mesh. It can transfer data across other mediums by putting C-Net in Altmode. If packets are addressed to the originating machine, they are directed back to the originating machine from this layer.

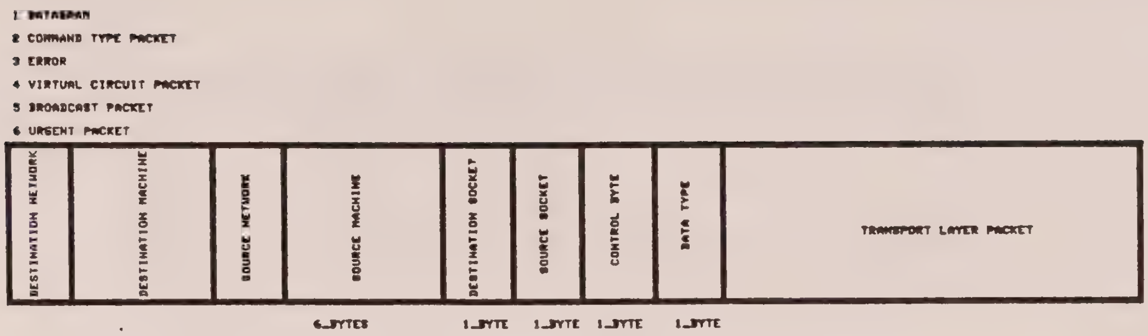

Figure 8: NETWORK LAYER PACKET

TRANSPORT LAYER SUPPORT

The transport layer is responsible for the flow control of the data. This layer supports two kind of services, **datagram** and **virtual circuits.** The network layer and data link layer of C-Net only support datagram service.

Datagram

Packets reach this layer from both higher and lower layers. Those packets marked as datagram or broadcast pass through this layer with minor alterations, such as the stripping of extra information which the higher layer does not need, or the adding of extra information needed by lower layers. After the packet leaves this layer, no information regarding the packet is retained here.

Virtual Circuits

This part of the transport layer can establish multiple virtual channels with transport layers of other machines on the network. A stream of data to a virtual channel is converted to datagram packets, which are buffered and then transferred to the network layer. The control flow section of the virtual circuit, using a sliding window algorithm, guarantees a reliable full-duplex communication between the transport layers. The packets that are lost on transmit are repeated, and the packets that are not received are requested. The packets are sequenced so the data coming out of a virtual circuit is in the same order in which it entered. An interrupt handler allows the interrupt packets to go to the head of the queue.

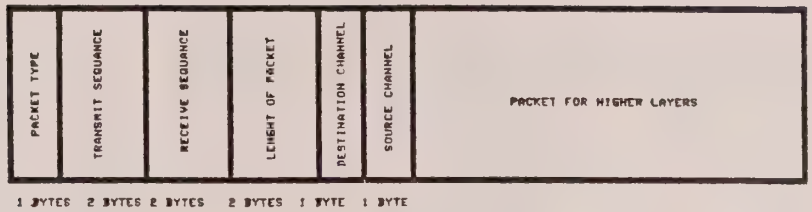

Figure 9: Virtual Circuit Packet

APPLICATION AND NETWORK INTERFACE

The following subroutines interface an application program or a terminal to the Session Layer of the network. These subroutines may later be integrated as part of the operating system.

snddatagram: Send a datagram packet through a specific socket to the session layer and, from there, to the network.

rcvpacket: Receive a packet through a specific socket.

sndpkt: Send a packet through a specific socket for a specific virtual channel.

chkrcvpkt: Check socket. If more packets can be received, receive them; if not, return and do not wait.

chksndpkt: Check socket. If more packets can be sent, send them; if not, return and do not wait.

closechn: Close a specific channel through a specific socket.

closesckt: Close specific socket, so network knows nobody is listening.

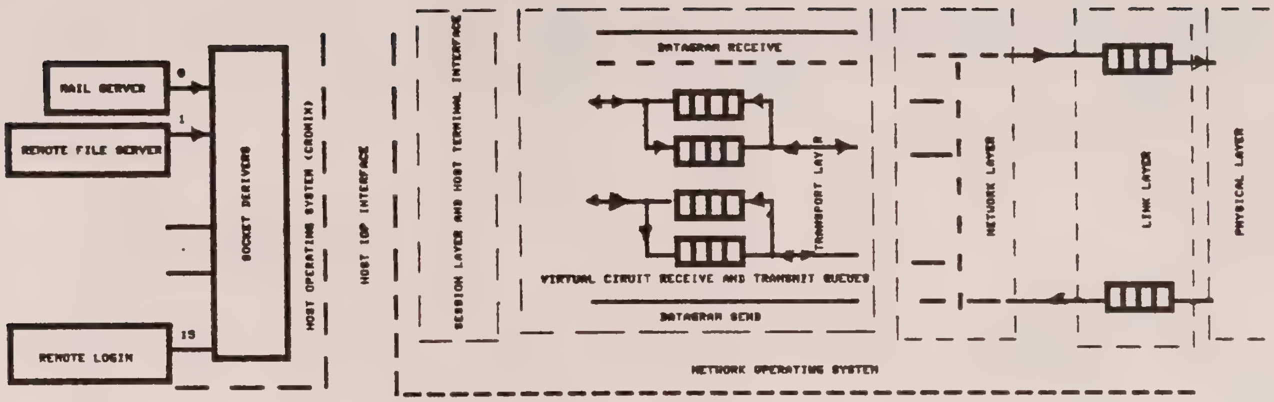

Figure 10: NETWORK ARCHITECTURE

C-NET APPLICATION LAYER

Once computers on the network were interconnected and different layers of communication protocols were designed and implemented, every computer on the network could exchange information with any other computer on the network. The following application utilities programs were written on top of other layers to make this exchange of data easier for the user.

Mail

This utility sends <u>files</u> and <u>messages</u> to the local machine or to other remote machines. The syntax is:

```
mail [file-list -to ] [destination list]
```

Example:

```
mail reportl memo* -to eng!roger r&d!joe joyce

message: i have sent the file reportl and all
the memos to roger at engineering machine,
joe at r&d machine, and joyce at local machine
```

Talk

This utility enables a user to have an online conversation with another user on a local or remote machine.

Example:

```
talk eng!roger
```

Rmail

This utility allows users to read mail sent to them on a remote machine. This utility requires that you know your password on the remote machine.

Example:

```
rmail eng
passwd:xxxxxx
```

This reads your mail from the machine eng (engineering).

Rbult

The Rbult utility enables users to read the bulletin board, located on a remote system.

Example:

```
rbult mailmastermachine
```

Broad

This utility sends a one-line message to all logged-on C-Net users, or to all logged-on users of a specific machine.

Example:

```
broad
There will be an earthquake in 10 seconds.
```

Remotelogin

This utility enables you to log in on a remote machine.

Example:

```
remotelogin eng!tom
passwd:xxxxxx

Logged in as tom to the engineering machine.
```

This utility is not yet completely implemented.

BIBLIOGRAPHY

1. Kenneth T. Pogran, David P. Reed, and David D. Clark, "An Introduction to Local Area Networks," Proceedings of the IEEE, pp. 16-36, November 1978.

2. Curt Terwilliger, "C-Bus, IOP, and Quadart," Cromemco Inc., I/O News, pp. 1-25, November 1980.

3. Digital, Intel, and Xerox, "The Ethernet -- a Local Area Network," Version 1.0, September 30, 1980.

4. IBM, "IBM Synchronous Data Link Control General Information," File No. Genl-09, IBM System Communications Division, March 1979.

5. John F. Shoch, Yogen K. Dalal, David Redell, and Ron C. Crane, "Evolution of the Ethernet Local Computer Network," IEEE Computer, pp. 10-27, August 1982.

6. Cromemco Inc., Cromemco CNI Instruction Manual, Cromemco part number 023-2019, April 1983.

7. Cromemco Inc., Cromemco C-Net User Manual, Cromemco part number 023-4059, September 1983.